A Probabilistic Corpus-Driven Model for Lexical-Functional Analysis

Rens Bod
Department of Computational Linguistics
University of Amsterdam
Spuistraat 134, NL-1012 VB Amsterdam
rens.bod@let.uva.nl

Ronald Kaplan
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, California 94304
kaplan@parc.xerox.com

Abstract

We develop a Data-Oriented Parsing (DOP) model based on the syntactic representations of Lexical-Functional Grammar (LFG). We start by summarizing the original DOP model for tree representations and then show how it can be extended with corresponding functional structures. The resulting LFG-DOP model triggers a new, corpus-based notion of grammaticality, and its probability models exhibit interesting behavior with respect to specificity and the interpretation of ill-formed strings.

1. Introduction

Data-Oriented Parsing (DOP) models of natural language embody the assumption that human language perception and production works with representations of past language experiences, rather than with abstract grammar rules (cf. Bod 1992, 95; Scha 1992; Sima'an 1995; Rajman 1995). DOP models therefore maintain large corpora of linguistic representations of previously occurring utterances. New utterances are analyzed by combining (arbitrarily large) fragments from the corpus; the occurrence-frequencies of the fragments are used to determine which analysis is the most probable one. In accordance with the general DOP architecture outlined by Bod (1995), a particular DOP model is described by specifying settings for the following four parameters:

- a formal definition of a well-formed *representation for utterance analyses*,
- a set of *decomposition operations* that divide a given utterance analysis into a set of fragments,
- a set of *composition operations* by which such fragments may be recombined to derive an analysis of a new utterance, and
- a definition of a *probability model* that indicates how the probability of a new utterance analysis is computed on the basis of the probabilities of the fragments that combine to make it up.

Previous instantiations of the DOP architecture were based on utterance-analyses represented as surface phrase-structure trees ("Tree-DOP", e.g. Bod 1993; Rajman 1995; Sima'an 1995; Goodman 1996; Bonnema et al. 1997). Tree-DOP uses two decomposition operations that produce connected subtrees of utterance representations: (1) the *Root* operation selects any node of a tree to be the root of the new subtree and erases all nodes except the selected node and the nodes it dominates; (2) the *Frontier* operation then chooses a set (possibly empty) of nodes in the new subtree different from its root and erases all subtrees dominated by the chosen nodes. The only composition operation used by Tree-DOP is a node-substitution operation that replaces the left-most nonterminal frontier node in a subtree with a fragment whose root category matches the category of the frontier node. Thus Tree-DOP provides tree-representations for new utterances by combining fragments from a corpus of phrase structure trees.

A Tree-DOP representation R can typically be derived in many different ways. If each derivation D has a probability $P(D)$, then the probability of deriving R is the sum of the individual derivation probabilities:

$$P(R) = \Sigma_{D \text{ derives } R} P(D)$$

A Tree-DOP derivation $D = \langle t_1, t_2 \ldots t_k \rangle$ is produced by a stochastic branching process. It starts by randomly choosing a fragment t_1 labeled with the initial category (e.g. S). At each subsequent step, a next fragment is chosen at random from among the set of competitors for composition into the current subtree. The process stops when a tree results with no nonterminal leaves. Let $CP(t \mid CS)$ denote the probability of choosing a tree t from a competition set CS containing t. Then the probability of a derivation is

$$P(\langle t_1, t_2 \ldots t_k \rangle) = \Pi_i CP(t_i \mid CS_i)$$

where the competition probability $CP(t \mid CS)$ is given by

$$CP(t \mid CS) = P(t) / \Sigma_{t' \in CS} P(t')$$

Here, $P(t)$ is the fragment probability for t in a given corpus. Let $T_{i-1} = t_1 \circ t_2 \circ \ldots \circ t_{i-1}$ be the subanalysis just before the ith step of the process, let $LNC(T_{i-1})$ denote the category of the leftmost nonterminal of T_{i-1}, and let $r(t)$ denote the root category of a fragment t. Then the competition set at the i^{th} step is

$$CS_i = \{ t : r(t) = LNC(T_{i-1}) \}$$

That is, the competition sets for Tree-DOP are determined by the category of the leftmost nonterminal of the current subanalysis. This is not the only possible definition of competition set. As Manning and Carpenter (1997) have shown, the competition sets can be made dependent on the composition operation. Their left-corner language model would also apply to Tree-DOP, yielding a different definition for the competition sets. But the properties of such Tree-DOP models have not been investigated.

Experiments with Tree-DOP on the Penn Treebank and the OVIS corpus show a consistent increase in parse accuracy when larger and more complex subtrees are taken into account (cf. Bod 1993, 95, 98; Bonnema et al. 1997; Sekine & Grishman 1995; Sima'an 1995). However, Tree-DOP is limited in that it cannot account for underlying syntactic (and semantic) dependencies that are not

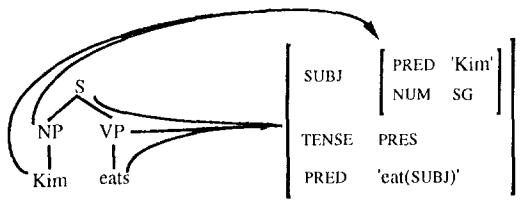
reflected directly in a surface tree. All modern linguistic theories propose more articulated representations and mechanisms in order to characterize such linguistic phenomena. DOP models for a number of richer representations have been explored (van den Berg et al. 1994; Tugwell 1995), but these approaches have remained context-free in their generative power. In contrast, Lexical-Functional Grammar (Kaplan & Bresnan 1982; Kaplan 1989), which assigns representations consisting of a surface constituent tree enriched with a corresponding functional structure, is known to be beyond context-free. In the current work, we develop a DOP model based on representations defined by LFG theory ("LFG-DOP"). That is, we provide a new instantiation for the four parameters of the DOP architecture. We will see that this basic LFG-DOP model triggers a new, corpus-based notion of grammaticality, and that it leads to a different class of its probability models which exhibit interesting properties with respect to specificity and the interpretation of ill-formed strings.

2. A DOP model based on Lexical-Functional representations

Representations

The definition of a well-formed representation for utterance-analyses follows from LFG theory, that is, every utterance is annotated with a c-structure, an f-structure and a mapping ϕ between them. The c-structure is a tree that describes the surface constituent structure of an utterance; the f-structure is an attribute-value matrix marking the grammatical relations of subject, predicate and object, as well as providing agreement features and semantic forms; and ϕ is a correspondence function that maps nodes of the c-structure into units of the f-structure (Kaplan & Bresnan 1982; Kaplan 1989). The following figure shows a representation for the utterance *Kim eats*. (We leave out some features to keep the example simple.)

(1)



Note that the ϕ correspondence function gives an explicit characterization of the relation between the superficial and underlying syntactic properties of an utterance, indicating how certain parts of the string carry information about particular units of underlying structure. As such, it will play a crucial role in our definition for the decomposition and composition operations of LFG-DOP. In (1) we see for instance that the NP node maps to the subject f-structure, and the S and VP nodes map to the outermost f-structure.

It is generally the case that the nodes in a subtree carry information only about the f-structure units that the subtree's root gives access to. The notion of accessibility is made precise in the following definition:

> An f-structure unit f is ϕ-*accessible* from a node n iff either n is ϕ-linked to f (that is, $f = \phi(n)$) or f is contained within $\phi(n)$ (that is, there is a chain of attributes that leads from $\phi(n)$ to f).

All the f-structure units in (1) are ϕ-accessible from for instance the S node and the VP node, but the TENSE and top-level PRED are not ϕ-accessible from the NP node.

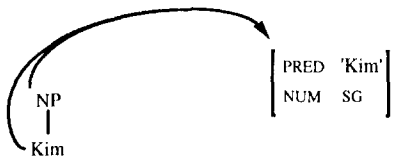
According to LFG theory, c-structures and f-structures must satisfy certain formal well-formedness conditions. A c-structure/f-structure pair is a *valid* LFG representation only if it satisfies the Non-branching Dominance, Uniqueness, Coherence and Completeness conditions (Kaplan & Bresnan 1982). Nonbranching Dominance demands that no c-structure category appears twice in a nonbranching dominance chain; Uniqueness asserts that there can be at most one value for any attribute in the f-structure; Coherence prohibits the appearance of grammatical functions that are not governed by the lexical predicate; and Completeness requires that all the functions that a predicate governs appear as attributes in the local f-structure.

Decomposition operations

Many different DOP models are compatible with the system of LFG representations. In this paper we outline a basic LFG-DOP model which extends the operations of Tree-DOP to take correspondences and f-structure features into account. The decomposition operations for this model will produce fragments of the composite LFG representations. These will consist of connected subtrees whose nodes are in ϕ-correspondence with sub-units of f-structures. We extend the *Root* and *Frontier* decomposition operations of Tree-DOP so that they also apply to the nodes of the c-structure while respecting the fundamental principles of c-structure/f-structure correspondence.

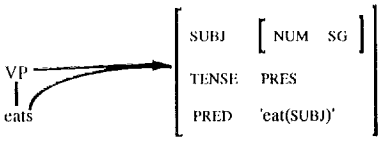
When a node is selected by the *Root* operation, all nodes outside of that node's subtree are erased, just as in Tree-DOP. Further, for LFG-DOP, all ϕ links leaving the erased nodes are removed and all f-structure units that are not ϕ-accessible from the remaining nodes are erased. *Root* thus maintains the intuitive correlation between nodes and the information in their corresponding f-structures. For example, if *Root* selects the NP in (1), then the f-structure corresponding to the S node is erased, giving (2) as a possible fragment:

(2)



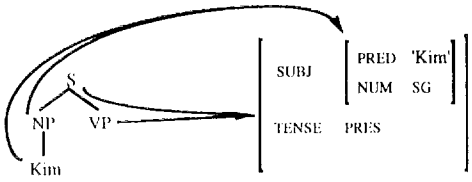
In addition the *Root* operation deletes from the remaining f-structure all semantic forms that are local to f-structures that correspond to erased c-structure nodes, and it thereby also maintains the fundamental two-way connection between words and meanings. Thus, if *Root* selects the VP node so that the NP is erased, the subject semantic form "Kim" is also deleted:

(3)



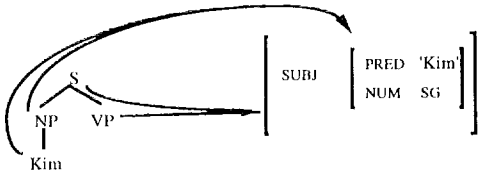
As with Tree-DOP, the *Frontier* operation then selects a set of frontier nodes and deletes all subtrees they dominate. Like *Root*, it also removes the ϕ links of the deleted nodes and erases any semantic form that corresponds to any of those nodes. *Frontier* does not delete any other f-structure features. This reflects the fact that all features are ϕ-accessible from the fragment's root even when nodes below the frontier are erased. For instance, if the VP in (1) is selected as a frontier node, *Frontier* erases the predicate "eat(SUBJ)" from the fragment:

(4)



Note that the *Root* and *Frontier* operations retain the subject's NUM feature in the VP-rooted fragment (3), even though the subject NP is not present. This reflects the fact, usually encoded in particular grammar rules or lexical entries, that verbs of English carry agreement features for their subjects. On the other hand, fragment (4) retains the predicate's TENSE feature, reflecting the possibility that English subjects might also carry information about their predicate's tense. Subject-tense agreement as encoded in (4) is a pattern seen in some languages (e.g. the split-ergativity pattern of languages like Hindi, Urdu and Georgian) and thus there is no universal principle by which fragments such as (4) can be ruled out. But in order to represent directly the possibility that subject-tense agreement is not a dependency of English, we also allow an S fragment in which the TENSE feature is deleted, as in (5).

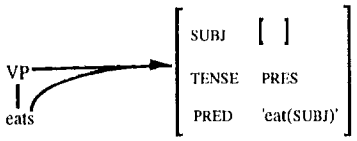
(5)



Fragment (5) is produced by a third decomposition operation, *Discard*, defined to construct generalizations of the fragments supplied by *Root* and *Frontier*. *Discard* acts to delete combinations of attribute-value pairs subject to the following restriction: *Discard* does not delete pairs whose values ϕ-correspond to remaining c-structure nodes.

This condition maintains the essential correspondences of LFG representations: if a c-structure and an f-structure are paired in one fragment provided by *Root* and *Frontier*, then *Discard* also pairs that c-structure with all generalizations of that fragment's f-structure. Fragment (5) results from applying *Discard* to the TENSE feature in (4). *Discard* also produces fragments such as (6), where the subject's number in (3) has been deleted:

(6)



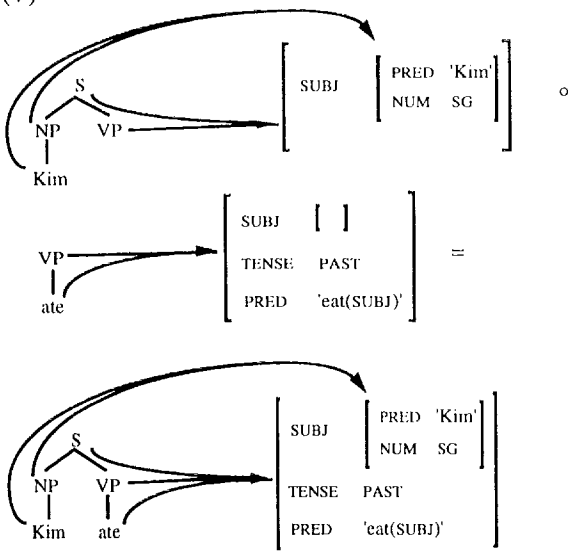
Again, since we have no language-specific knowledge apart from the corpus, we have no basis for ruling out fragments like (6). Indeed, it is quite intuitive to omit the subject's number in fragments derived from sentences with past-tense verbs or modals. Thus the specification of *Discard* reflects the fact that LFG representations, unlike LFG grammars, do not indicate unambiguously the c-structure source (or sources) of their f-structure feature values.

The composition operation

In LFG-DOP the operation for combining fragments, again indicated by $\circ$, is carried out in two steps. First the c-structures are combined by left-most substitution subject to the category-matching condition, just as in Tree-DOP. This is followed by the recursive unification of the f-structures corresponding to the matching nodes. The result retains the ϕ correspondences of the fragments being combined. A derivation for an LFG-DOP representation R is a sequence of fragments the first of which is labeled with S and for which the iterative application of the composition operation produces R.

We show in (7) the effect of the LFG composition operation using two fragments from representations of an imaginary corpus containing the sentences *Kim eats* and *People ate*. The VP-rooted fragment is substituted for the VP in the first fragment, and the second f-structure unifies with the first f-structure, resulting in a representation for the new sentence *Kim ate*.

(7)

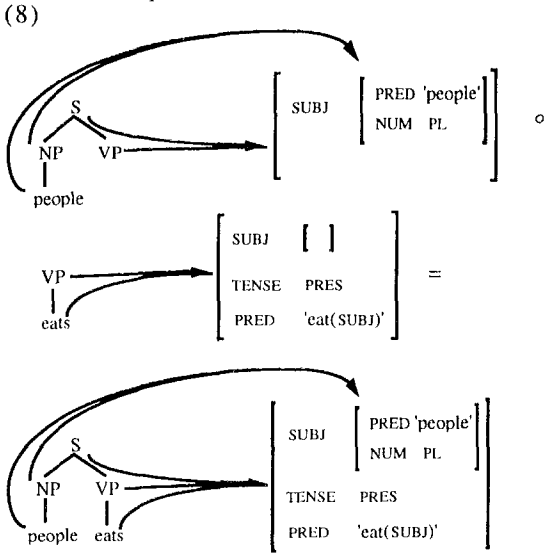


This representation satisfies the well-formedness conditions and is therefore valid. Note that in LFG-DOP, as in Tree-DOP, the same representation may be produced by several derivations involving different fragments.

Another valid representation for the sentence *Kim ate* could be composed from a fragment for *Kim* that does not preserve the number feature, leading to a representation which is unmarked for number. The probability models we discuss below have the desirable property that they tend to assign higher probabilities to more specific representations.

The following derivation produces a valid representation for the intuitively ungrammatical sentence *People eats*:

(8)



This system of fragments and composition thus provides a representational basis for a robust model of language comprehension in that it assigns at least some representations to many strings that would generally be regarded as ill-formed. A correlate of this advantage, however, is the fact that it does not offer a direct formal account of metalinguistic judgments of grammaticality. Nevertheless, we can reconstruct the notion of grammaticality by means of the following definition:

> A sentence is *grammatical with respect to a corpus* if and only if it has at least one valid representation with at least one derivation whose fragments are produced only by *Root* and *Frontier* and not by *Discard*.

Thus the system is robust in that it assigns three representations (singular, plural, and unmarked as the subject's number) to the string *People eats*, based on fragments for which the number feature of *people*, *eats*, or both has been discarded. But unless the corpus contains non-plural instances of *people* or non-singular instances of *eats*, there will be no *Discard*-free derivation and the string will be classified as ungrammatical (with respect to the corpus).

Probability models

As in Tree-DOP, an LFG-DOP representation R can typically be derived in many different ways. If each derivation D has a probability $P(D)$, then the probability of deriving R is again the probability of producing it by any of its derivations. This is the sum of the individual derivation probabilities:

(9) $\quad P(R) = \Sigma_{D \text{ derives } R} P(D)$

An LFG-DOP derivation is also produced by a stochastic branching process which at each step makes a random selection from a competition set of competing fragments. Let $CP(f \mid CS)$ denote the probability of choosing a fragment f from a competition set CS containing f, then the probability of a derivation $D = \langle f_1, f_2 \ldots f_k \rangle$ is

(10) $\quad P(\langle f_1, f_2 \ldots f_k \rangle) = \Pi_i CP(f_i \mid CS_i)$

where as in Tree-DOP, $CP(f \mid CS)$ is expressed in terms of fragment probabilities $P(f)$ by the formula

(11) $\quad CP(f \mid CS) = P(f) / \Sigma_{f' \in CS} P(f')$

Tree-DOP is the special case where there are no conditions of validity other than the ones that are enforced at each step of the stochastic process by the composition operation. This is not generally the case and is certainly not the case for the Completeness Condition of LFG representations: Completeness is a property of a final representation that cannot be evaluated at any intermediate steps of the process. However, we can define probabilities for the valid representations by sampling only from such representations in the output of the stochastic process. The probability of sampling a particular valid representation R is given by

(12) $\quad P(R \mid R \text{ is valid}) = P(R) / \Sigma_{R' \text{ is valid}} P(R')$

This formula assigns probabilities to valid representations whether or not the stochastic process guarantees validity. The valid representions for a particular utterance u are obtained by a further sampling step and their probabilities are given by:

(13) $\quad P(R \mid R \text{ is valid and yields } u) =$
$\qquad P(R) / \Sigma_{R' \text{ is valid and yields } u} P(R')$

The formulas (9) through (13) will be part of any LFG-DOP probability model. The models will differ only in how the competition sets are defined, and this in turn depends on which well-formedness conditions are enforced on-line during the stochastic branching process and which are evaluated by the off-line validity sampling process.

One model, which we call M1, is a straightforward extension of Tree-DOP's probability model. This computes the competition sets only on the basis of the category-matching condition, leaving all other well-formedness conditions for off-line sampling. Thus for M1 the competition sets are defined simply in terms of the categories of a fragment's c-structure root node. Suppose that $F_{i-1} = f_1 \circ f_2 \circ \ldots \circ f_{i-1}$ is the current subanalysis at the beginning of step i in the process, that $LNC(F_{i-1})$ denotes the category of the leftmost nonterminal node of the c-structure of F_{i-1}, and that $r(f)$ is now interpreted as the root-node category of f's c-structure component. Then the competition set for the i^{th} step is

(14) $\quad CS_i = \{ f : r(f) = LNC(F_{i-1}) \}$

Since these competition sets depend only on the category of the leftmost nonterminal of the current c-structure, the competition sets group together all fragments with the same root category, independent of any other properties they may have or that a particular derivation may have. The competition

probability for a fragment can be expressed by the formula

(15) $CP(f) = P(f) / \Sigma_{f'\,:\,r(f')=r(f)} P(f')$

We see that the choice of a fragment at a particular step in the stochastic process depends only on the category of its root node; other well-formedness properties of the representation are not used in making fragment selections. Thus, with this model the stochastic process may produce many invalid representations; we rely on sampling of valid representations and the conditional probabilities given by (12) and (13) to take the Uniqueness, Coherence, and Completeness Conditions into account.

Another possible model (M2) defines the competition sets so that they take a second condition, Uniqueness, into account in addition to the root node category. For M2 the competing fragments at a particular step in the stochastic derivation process are those whose c-structures have the same root node category as $LNC(F_{i-1})$ and also whose f-structures are consistently unifiable with the f-structure of F_{i-1}. Thus the competition set for the i^{th} step is

(16) $CS_i = \{ f : r(f)=LNC(F_{i-1}) \text{ and } f \text{ is unifiable with the f-structure of } F_{i-1} \}$

Although it is still the case that the category-matching condition is independent of the derivation, the unifiability requirement means that the competition sets vary according to the representation produced by the sequence of previous steps in the stochastic process. Unifiability must be determined at each step in the process to produce a new competition set, and the competition probability remains dependent on the particular step:

(17) $CP(f_i \mid CS_i) =$
$P(f_i) / \Sigma_{f'\,:\,r(f')=r(f_i) \text{ and } f' \text{ is unifiable with } F_{i-1}} P(f')$

On this model we again rely on sampling and the conditional probabilities (12) and (13) to take just the Coherence and Completeness Conditions into account.

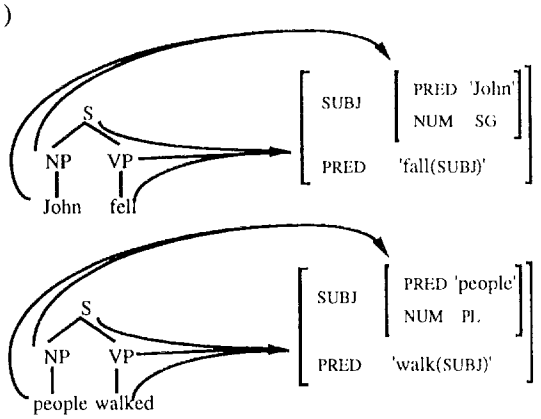
In model M3 we define the stochastic process to enforce three conditions, Coherence, Uniqueness and category-matching, so that it only produces representations with well-formed c-structures that correspond to coherent and consistent f-structures. The competition probabilities for this model are given by the obvious extension of (17). It is not possible, however, to construct a model in which the Completeness Condition is enforced during the derivation process. This is because the satisfiability of the Completeness Condition depends not only on the results of previous steps of a derivation but also on the following steps (see Kaplan & Bresnan 1982). This nonmonotonic property means that the appropriate step-wise competition sets cannot be defined and that this condition can only be enforced at the final stage of validity sampling.

In each of these three models the category-matching condition is evaluated on-line during the derivation process while other conditions are either evaluated on-line or off-line by the after-the-fact sampling process. LFG-DOP is crucially different from Tree-DOP in that at least one validity requirement, the Completeness Condition, must always be left to the post-derivation process. Note that a number of other models are possible which enforce other combinations of these three conditions.

3. Illustration and properties of LFG-DOP

We illustrate LFG-DOP using a very small corpus consisting of the two simplified LFG representations shown in (18):

(18)



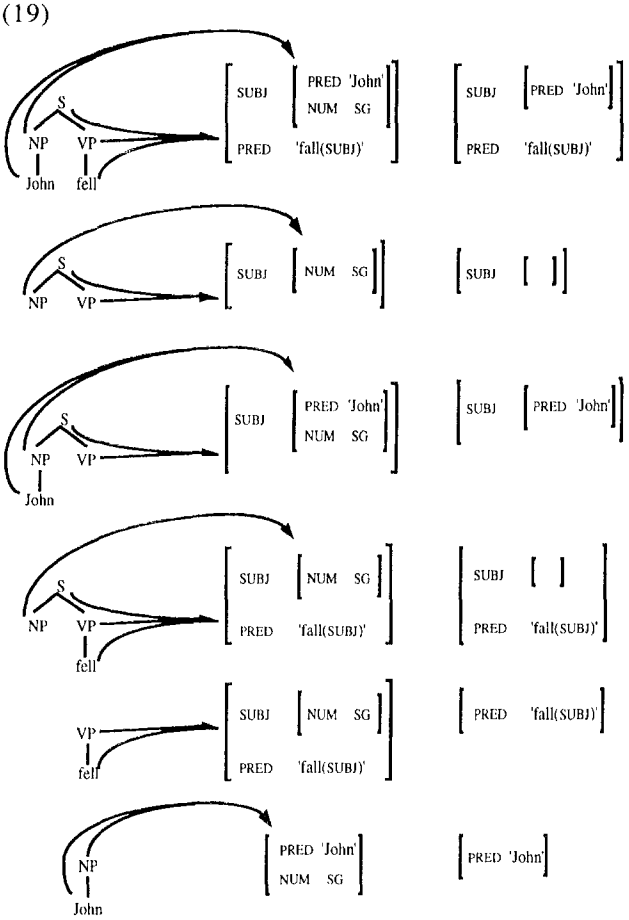
The fragments from this corpus can be composed to provide representations for the two observed sentences plus two new utterances, *John walked* and *People fell*. This is sufficient to demonstrate that the probability models M1 and M2 assign different probabilities to particular representations. We have omitted the TENSE feature and the lexical categories N and V to reduce the number of the fragments we have to deal with. Applying the *Root* and *Frontier* operators systematically to the first corpus representation produces the fragments in the first column of (19), while the second column shows the additional f-structure that is associated with each c-structure by the *Discard* operation.

A total of 12 fragments are produced from this representation, and by analogy 12 fragments with either PL or unmarked NUM values will also result from *People walked*. Note that the [S NP VP] fragment with the unspecified NUM value is produced for both sentences and thus its corpus frequency is 2. There are 14 other S-rooted fragments, 4 NP-rooted fragments, and 4 VP-rooted fragments; each of these occurs only once.

These fragments can be used to derive three different representations for *John walked* (singular, plural, and unmarked as the subject's number). To facilitate the presentation of our derivations and probability calculations, we denote each fragment by an abbreviated name that indicates its c-structure root-node category, the sequence of its frontier-node labels, and whether its subject's number is SG, PL, or unmarked (indicated by U). Thus the first fragment in (19) is referred to as S/John-fell/SG and the unmarked fragment that *Discard* produces from it is referred to as S/John-fell/U. Given this naming convention, we can specify one of the derivations for *John walked* by the expression S/NP-VP/U ∘ NP/John/SG ∘ VP/walked/U, corresponding to an analysis in which the subject's number is marked as SG. The fragment VP/walked/U of course comes from *People walked*,

the second corpus sentence, and does not appear in (19).

(19)



Model M1 evaluates only the Tree-DOP root-category condition during the stochastic branching process, and the competition sets are fixed independent of the derivation. The probability of choosing the fragment S/NP-VP/U, given that an S-rooted fragment is required, is always 2/16, its frequency divided by the sum of the frequencies of all the S fragments. Similarly, the probability of then choosing NP/John/SG to substitute at the NP frontier node is 1/4, since the NP competition set contains 4 fragments each with frequency 1. Thus, under model M1 the probability of producing the complete derivation S/NP-VP/U ∘ NP/John/SG ∘ VP/walked/U is 2/16×1/4×1/4=2/256. This probability is small because it indicates the likelihood of this derivation compared to other derivations for *John walked* and for the three other analyzable strings. The computation of the other M1 derivation probabilities for *John walked* is left to the reader. There are 5 different derivations for the representation with SG number and 5 for the PL number, while there are only 3 ways of producing the unmarked number U. The conditional probabilities for the particular representations (SG, PL, U) can be calculated by (9) and (13), and are given below.

P(NUM=SG | valid and yield = *John walked*) = .353
P(NUM=PL | valid and yield = *John walked*) = .353
P(NUM=U | valid and yield = *John walked*) = .294

We see that the two specific representations are equally likely and each of them is more probable than the representation with unmarked NUM.

Model M2 produces a slightly different distribution of probabilities. Under this model, the consistency requirement is used in addition to the root-category matching requirement to define the competition sets at each step of the branching process. This means that the first fragment that instantiates the NUM feature to either SG or PL constrains the competition sets for the following choices in a derivation. Thus, having chosen the NP/John/SG fragment in the derivation S/NP-VP/U ∘ NP/John/SG ∘ VP/walked/U, only 3 VP fragments instead of 4 remain in the competition set at the next step, since the VP/walked/PL fragment is no longer available. The probability for this derivation under model M2 is therefore 2/16×1/4×1/3=2/192, slightly higher than the probability assigned to it by M1. Table 1 shows the complete set of derivations and their M2 probabilities for *John walked.*

S/NP-VP/U ∘ NP/John/SG ∘ VP/walked/U	SG	2/16 x 1/4 x 1/3
S/NP-VP/SG ∘ NP/John/SG ∘ VP/walked/U	SG	1/16 x 1/3 x 1/3
S/NP-VP/SG ∘ NP/John/U ∘ VP/walked/U	SG	1/16 x 1/3 x 1/3
S/NP-walked/U ∘ NP/John/SG	SG	1/16 x 1/4
S/John-VP/SG ∘ VP/walked/U	SG	1/16 x 1/3
P(NUM=SG and yield = *John walked*) = 35/576 = .061		
P(NUM=SG | valid and yield = *John walked*) = 70/182 = .38		
S/NP-VP/U ∘ NP/John/U ∘ VP/walked/PL	PL	2/16 x 1/4 x 1/4
S/NP-VP/PL ∘ NP/John/U ∘ VP/walked/PL	PL	1/16 x 1/3 x 1/3
S/NP-VP/PL ∘ NP/John/U ∘ VP/walked/U	PL	1/16 x 1/3 x 1/3
S/NP-walked/PL ∘ NP/John/U	PL	1/16 x 1/3
S/John-VP/U ∘ VP/walked/PL	PL	1/16 x 1/4
P(NUM=PL and yield = *John walked*) = 33.5/576 = .058		
P(NUM=PL | valid and yield = *John walked*) = 67/182 = .37		
S/NP-VP/U ∘ NP/John/U ∘ VP/walked/U	U	2/16 x 1/4 x 1/4
S/NP-walked/U ∘ NP/John/U	U	1/16 x 1/4
S/John-VP/U ∘ VP/walked/U	U	1/16 x 1/4
P(NUM=U and yield = *John walked*) = 22.5/576 = .039		
P(NUM=U | valid and yield = *John walked*) = 45/182 = .25		

Table 1: Model M2 derivations, subject number features, and probabilities for *John walked*

The total probability for the derivations that produce *John walked* is .158, and the conditional probabilities for the three representations are:

P(NUM=SG | valid and yield = *John walked*) = .38
P(NUM=PL | valid and yield = *John walked*) = .37
P(NUM=U | valid and yield = *John walked*) = .25

For model M2 the unmarked representation is less likely than under M1, and now there is a slight bias in favor of the value SG over PL. The SG value is favored because it is carried by substitutions for the left-most word of the utterance and thus reduces competition for subsequent choices. The value PL would be more probable for the sentence *People fell.* Thus both models give higher probability to the more specific representations. Moreover, M1 assigns the same probability to SG and PL, whereas M2 doesn't.

M2 reflects a left-to-right bias (which might be psycholinguistically interesting -- a so-called primacy effect), whereas M1 is, like Tree-DOP, order independent.

It turns out that all LFG-DOP probability models (M1, M2 and M3) display a preference for the most specific representation. This preference partly depends on the number of derivations: specific representations tend to have more derivations than generalized (i.e., unmarked) representations, and consequently tend to get higher probabilities -- other things being equal. However, this preference also depends on the number of feature values: the more feature values, the longer the minimal derivation length must be in order to get a preference for the most specific representation (Cormons, forthcoming).

The bias in favor of more specific representations, and consequently fewer Discard-produced feature generalizations, is especially interesting for the interpretation of ill-formed input strings. Bod & Kaplan (1997) show that in analyzing an intuitively ungrammatical string like *These boys walks*, there is a probabilistic accumulation of evidence for the plural interpretation over the singular and unmarked one (for all models M1, M2 and M3). This is because both *These* and *boys* carry the PL feature while only *walks* is a source for the SG feature, leading to more derivations for the PL reading of *These boys walks*. In case of "equal evidence" as in the ill-formed string *Boys walks*, model M1 assigns the same probability to PL and SG, while models M2 and M3 prefer the PL interpretation due to their left-to-right bias.

4. Conclusion and computational issues

Previous DOP models were based on context-free tree representations that cannot adequately represent all linguistic phenomena. In this paper, we gave a DOP model based on the more articulated representations provided by LFG theory. LFG-DOP combines the advantages of two approaches: the linguistic adequacy of LFG together with the robustness of DOP. LFG-DOP triggers a new, corpus-based notion of grammaticality, and its probability models exhibit a preference for the most specific analysis containing the fewest number of feature generalizations.

The main goal of this paper was to provide the theoretical background of LFG-DOP. As to the computational aspects of LFG-DOP, the problem of finding the most probable representation of a sentence is NP-hard even for Tree-DOP. This problem may be tackled by Monte Carlo sampling techniques (as in Tree-DOP, cf. Bod 1995) or by computing the Viterbi n best derivations of a sentence. Other optimization heuristics may consist of restricting the fragment space, for example by putting an upper bound on the fragment depth, or by constraining the decomposition operations. To date, a couple of LFG-DOP implementations are either operational (Cormons, forthcoming) or under development, and corpora with LFG representations have recently been developed (at XRCE France and Xerox PARC). Experiments with these corpora will be presented in due time.

Acknowledgments

We thank Joan Bresnan, Mary Dalrymple, Mark Johnson, Martin Kay, John Maxwell, Remko Scha, Khalil Sima'an, Andy Way and three anonymous reviewers for helpful comments. We are most grateful to Boris Cormons whose comments were particularly helpful. This research was supported by NWO, the Dutch Organization for Scientific Research. The initial stages of this work were carried out while the second author was a Fellow of the Netherlands Institute for Advanced Study (NIAS). Subsequent stages were also carried out while the first author was a Consultant at Xerox PARC.

References

M. van den Berg, R. Bod and R. Scha 1994. "A Corpus-Based Approach to Semantic Interpretation", *Proceedings Ninth Amsterdam Colloquium*, Amsterdam, The Netherlands.

R. Bod 1992. "A Computational Model of Language Performance: Data Oriented Parsing", *Proceedings COLING-92*, Nantes, France.

R. Bod 1993. "Using an Annotated Corpus as a Stochastic Grammar", *Proceedings EACL'93*, Utrecht, The Netherlands.

R. Bod 1995. *Enriching Linguistics with Statistics: Performance Models of Natural Language*, ILLC Dissertation Series 1995-14, University of Amsterdam

R. Bod 1998. "Spoken Dialogue Interpretation with the DOP Model", this proceedings.

R. Bod and R. Kaplan 1997. "On Performance models for Lexical-Functional Analysis", Paper presented at the *Computational Psycholinguistics Conference* 1997, Berkeley (Ca).

R. Bonnema, R. Bod and R. Scha 1997. "A DOP Model for Semantic Interpretation", *Proceedings ACL/EACL-97*, Madrid, Spain.

B. Cormons, forthcoming. *Analyse et desambiguisation: Une approche purement à base de corpus (Data-Oriented Parsing) pour le formalisme des Grammaires Lexicales Fonctionnelles*, PhD thesis, Université de Rennes, France.

J. Goodman 1996. "Efficient Algorithms for Parsing the DOP Model", *Proceedings Empirical Methods in Natural Language Processing*, Philadelphia, Pennsylvania.

R. Kaplan 1989. "The Formal Architecture of Lexical-Functional Grammar", *Journal of Information Science and Engineering*, vol. 5, 305-322.

R. Kaplan and J. Bresnan 1982. "Lexical-Functional Grammar: A Formal System for Grammatical Representation", in J. Bresnan (ed.), *The Mental Representation of Grammatical Relations*, The MIT Press, Cambridge, MA.

C. Manning and B. Carpenter 1997. "Probabilistic parsing using left corner language models", *Proceedings IWPT'97*, Boston (Mass.).

M. Rajman 1995. "Approche Probabiliste de l'Analyse Syntaxique", *Traitement Automatique des Langues*, vol. 36(1-2).

R. Scha 1992. "Virtuele Grammatica's en Creatieve Algoritmen", *Gramma/TTT* 1(1).

S. Sekine and R. Grishman 1995. "A Corpus-based Probabilistic Grammar with Only Two Non-terminals", *Proceedings Fourth International Workshop on Parsing Technologies*, Prague, Czech Republic.

K. Sima'an 1995. "An optimized algorithm for Data Oriented Parsing", in R. Mitkov and N. Nicolov (eds.), *Recent Advances in Natural Language Processing 1995*, John Benjamins, Amsterdam.

D. Tugwell 1995. "A State-Transition Grammar for Data-Oriented Parsing", *Proceedings European Chapter of the ACL'95*, Dublin, Ireland.